

Jupyter Notebook + Google Colaboratory Instructions

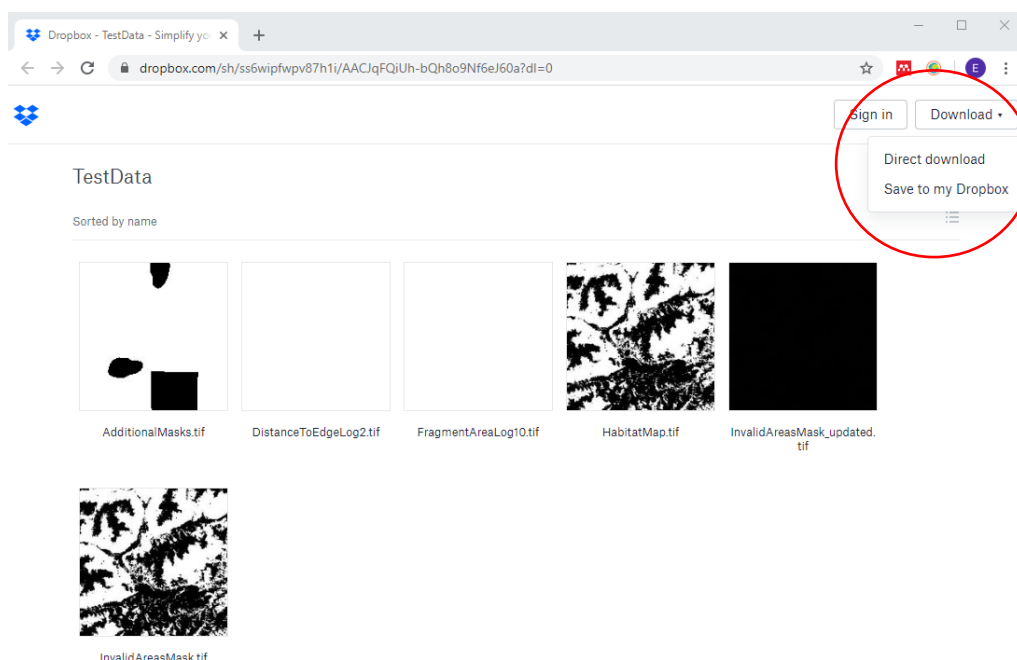
This document provides instructions for generating designs using Jupyter Notebook files hosted on Google Colaboratory. This allows you to run the code without the need to download or install any software. Please follow the instructions below.

Test Data

To follow the tutorial we have provided example files on dropbox. **Please download all files in the TestData folder, from the following link:**

<https://www.dropbox.com/sh/ss6wipfwpv87h1i/AACJqFQiUh-bQh8o9Nf6eJ60a?dl=0>

Once downloaded, please save and unzip the folder in your chosen directory.



We will see how these files are used as we go through the notebooks, briefly we have:

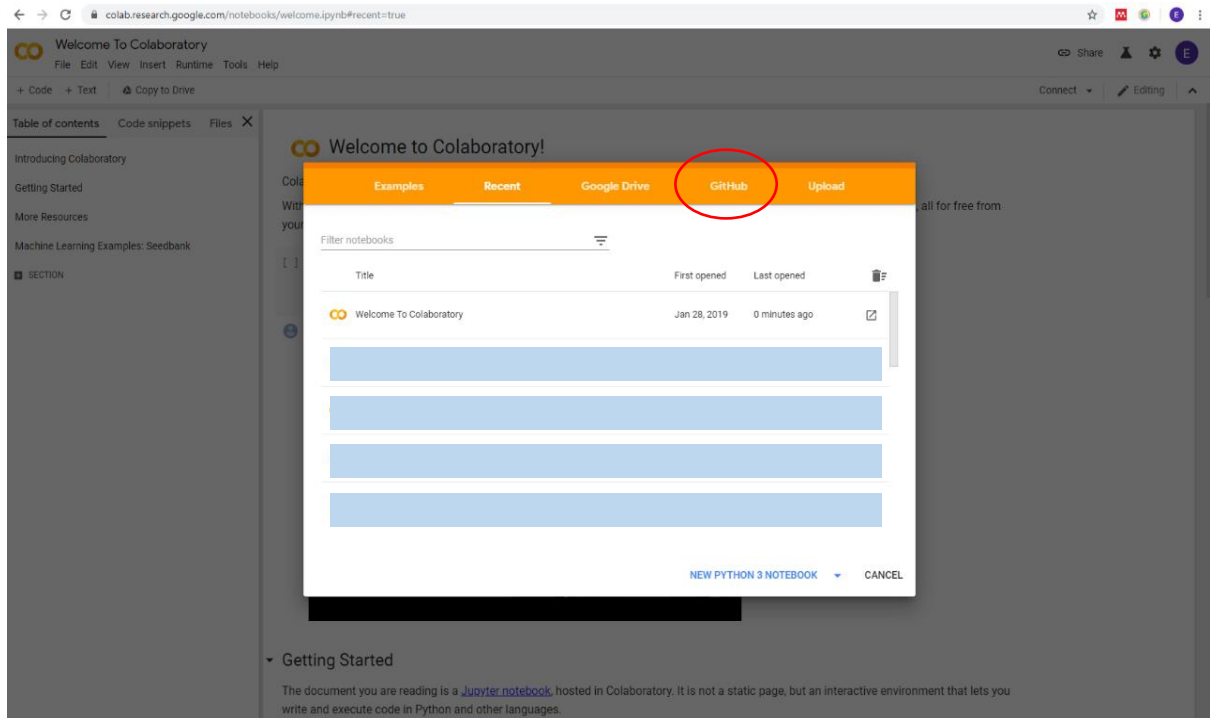
- **InvalidAreasMask.tif:** A binary map showing valid (coded with 1) and invalid (coded with 0) areas in the landscape
- **InvalidAreasMask_updated.tif:** An updated version of **InvalidAreasMask.tif**, with extra invalid areas added. This is used to demonstrate adapting designs
- **HabitatMap.tif:** A two class habitat map (with non-habitat = 0, habitat = 1). This was used to generated the two fragmentation metric maps:
 - **DistanceToEdgeLog2.tif:** A log2 scaled distance to nearest habitat edge map
 - **FragmentAreaLog10.tif:** A log10 scaled fragment area map

Please Note: All files are in georeferenced tiff format, and information on projection and resolution are used to output results in longitude/ latitude coordinates. This is important to know when inputting your own study site.

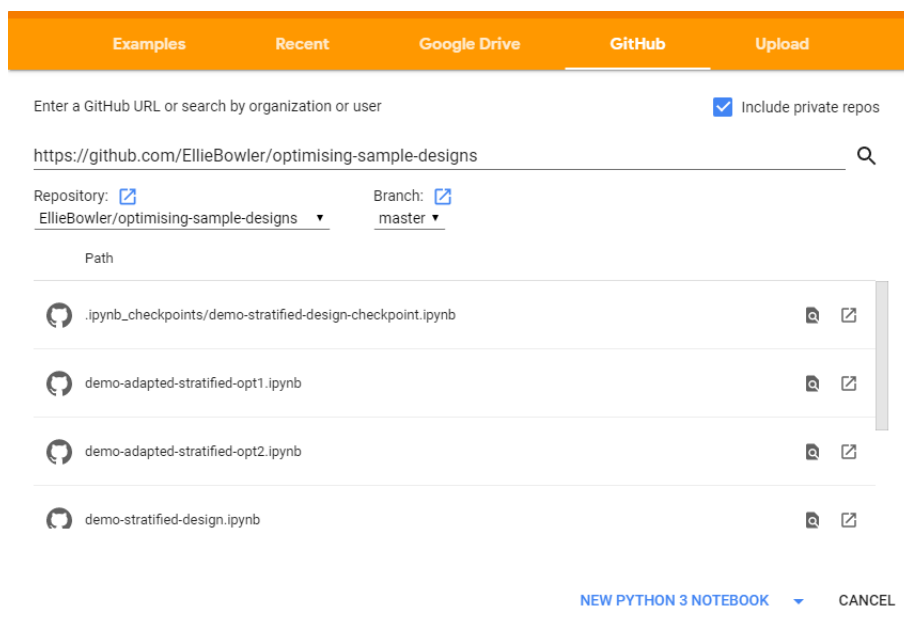
Google Colaboratory

Now we will open up our notebook files in Google Colaboratory, which we can use to run our python code online. **Please Note:** It is best to use Google Chrome browser.

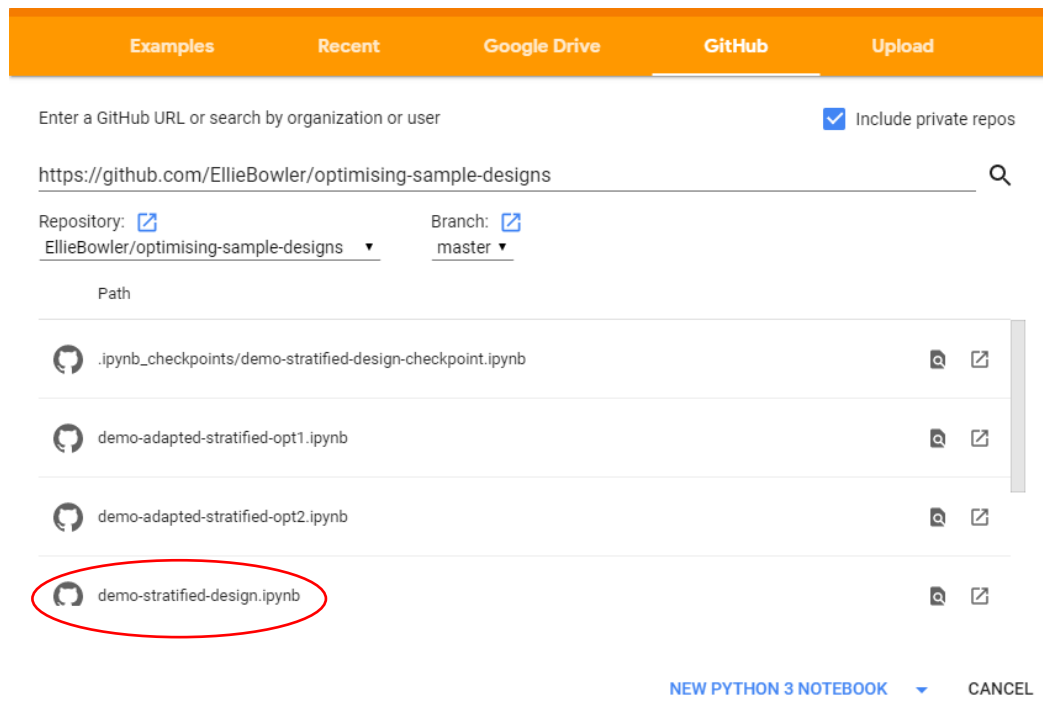
1. First open up Colab via this link: <https://colab.research.google.com/>
You should see the following pop up. Please click on the **GitHub tab**, circled in red.



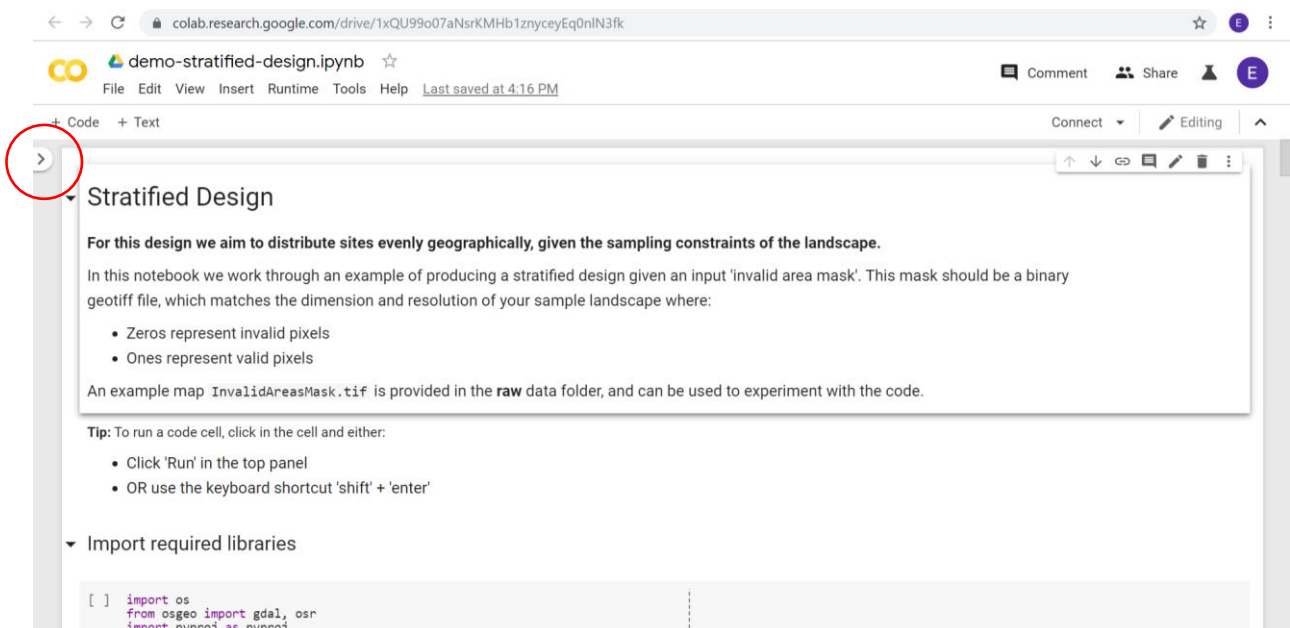
2. Under the top line 'Enter a GitHub URL or search by organization or user', enter the following github repo link <https://github.com/ElleBowler/optimising-sample-designs>. If this does not bring up the files, search by user instead with the name ElleBowler.



3. All jupyter notebooks (with file extension .ipynb) should appear listed as below. For this demonstration, **click on demo-stratified-design.ipynb**.



4. You should see the following page. This is the notebook for the Stratified Design. To run the code, we will need to **upload our test files**. To do this click on the arrow on the left of the notebook, circled in red.



5. A panel will open with 'Table of Contents', 'Code Snippets' and 'Files'. Click on Files.

The screenshot shows the Google Colaboratory interface for a notebook titled 'demo-stratified-design.ipynb'. The top menu bar includes 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. Below the menu, there are tabs for '+ Code', '+ Text', and 'Copy to Drive'. The left sidebar contains a 'Table of contents' panel with 'Table of contents', 'Code snippets', and 'Files' tabs. The 'Files' tab is active, showing a file explorer with a 'sample_data' folder. The main area displays the 'Stratified Design' notebook content, which includes a description of the design goal, a list of invalid pixels, and a code cell for importing libraries and defining a function.

demo-stratified-design.ipynb

File Edit View Insert Runtime Tools Help Unsaved changes since 12:18 PM

+ Code + Text Copy to Drive

Table of contents Code snippets Files X

UPLOAD REFRESH MOUNT DRIVE

sample_data

Stratified Design

For this design we aim to distribute sites evenly geographically, given the sampling constraints of the landscape.

In this notebook we work through an example of producing a stratified design given an input 'invalid area mask'. This mask should be a binary geotiff file, which matches the dimension and resolution of your sample landscape where:

- Zeros represent invalid pixels
- Ones represent valid pixels

An example map `InvalidAreasMask.tif` is provided in the `raw` data folder, and can be used to experiment with the code.

Tip: To run a code cell, click in the cell and either:

- Click 'Run' in the top panel
- OR use the keyboard shortcut 'shift' + 'enter'

Import required libraries

```
[ ] import os
from osgeo import gdal, osr
import numpy as np
from random import randint
from scipy import ndimage
from copy import copy
import time
import pandas as pd
from matplotlib import pyplot as plt
%matplotlib inline

def get_file_info(file_path):
    """
    Function which extracts a geotiff file as numpy array, and saves geographic projection information
    input:
        file_path: (str) Path to the file
    output:
        file_map: (np.array) The extracted map
        nbins: (int) The number of categories in the map
        res: (float) Resolution of the map in meters
```

6. You can now upload your files to Google Colaboratory, either by using the 'UPLOAD' button, or by dragging and dropping. Please copy all files across, and ensure they are in the same directory as the sample_data folder as shown below.

The screenshot shows the Google Colaboratory interface for the same notebook. The 'Files' panel on the left is now expanded, showing a list of files to be uploaded. The 'UPLOAD' button is highlighted with a red box. The list of files includes 'sample_data', 'AdditionalMasks.tif', 'DistanceToEdgeLog2.tif', 'FragmentAreaLog10.tif', 'HabitatMap.tif', 'InvalidAreasMask.tif', and 'InvalidAreasMask_updated.tif'. The main area of the notebook remains the same, showing the 'Stratified Design' content.

demo-stratified-design.ipynb

File Edit View Insert Runtime Tools Help Unsaved changes since 12:18 PM

+ Code + Text Copy to Drive

Table of contents Code snippets Files X

UPLOAD REFRESH MOUNT DRIVE

sample_data

AdditionalMasks.tif

DistanceToEdgeLog2.tif

FragmentAreaLog10.tif

HabitatMap.tif

InvalidAreasMask.tif

InvalidAreasMask_updated.tif

Stratified Design

For this design we aim to distribute sites evenly geographically, given the sampling constraints of the landscape.

In this notebook we work through an example of producing a stratified design given an input 'invalid area mask'. This mask should be a binary geotiff file, which matches the dimension and resolution of your sample landscape where:

- Zeros represent invalid pixels
- Ones represent valid pixels

An example map `InvalidAreasMask.tif` is provided in the `raw` data folder, and can be used to experiment with the code.

Tip: To run a code cell, click in the cell and either:

- Click 'Run' in the top panel
- OR use the keyboard shortcut 'shift' + 'enter'

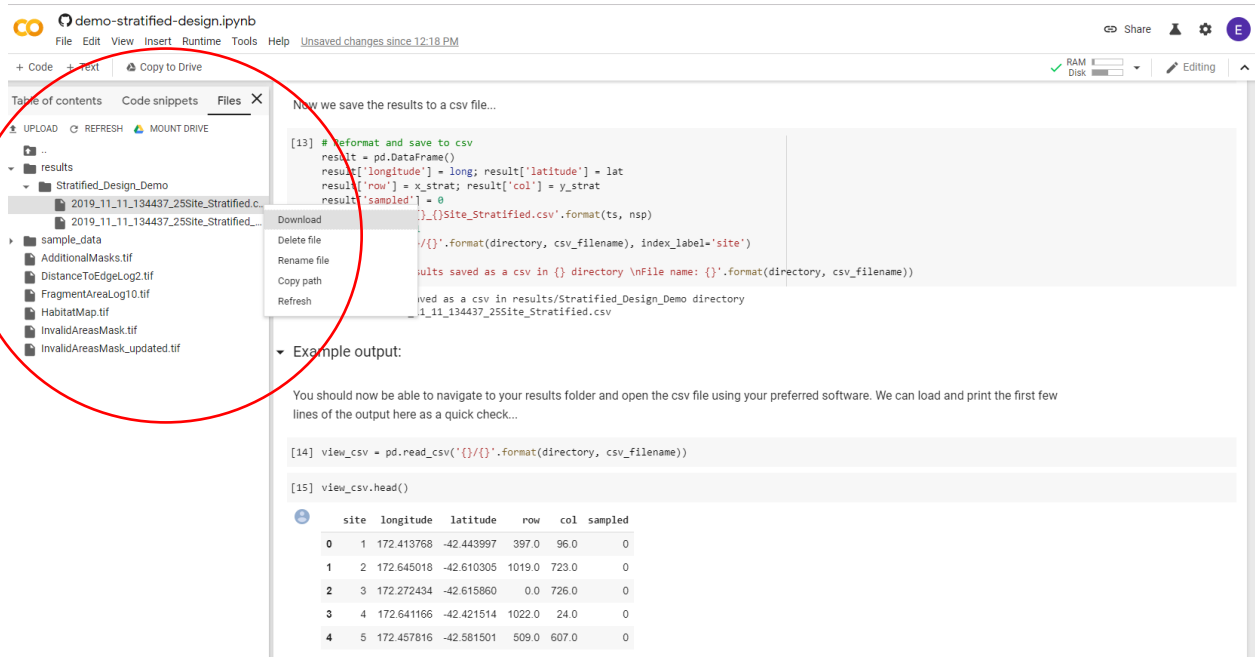
Import required libraries

```
[ ] import os
from osgeo import gdal, osr
import numpy as np
from random import randint
from scipy import ndimage
from copy import copy
import time
import pandas as pd
from matplotlib import pyplot as plt
%matplotlib inline

def get_file_info(file_path):
    """
    Function which extracts a geotiff file as numpy array, and saves geographic projection information
    input:
        file_path: (str) Path to the file
    output:
        file_map: (np.array) The extracted map
        nbins: (int) The number of categories in the map
        res: (float) Resolution of the map in meters
```

7. You should now be able to run through the notebook!

Instructions are provided within the notebook, please run through each block sequentially. Your results will be generated and saved in a **results** folder, shown in the file panel. You can download these directly from colabotatory and view them in your chosen software.



The screenshot shows a Jupyter Notebook interface with a file panel on the left and a code editor on the right. A red circle highlights the file panel, specifically the 'results' folder and its contents. The code editor shows a Python script that saves results to a CSV file and displays the first few rows of the output.

File panel contents (under 'results' folder):

- Stratified_Design_Demo
- 2019_11_11_134437_25Site_Stratified.c...
- 2019_11_11_134437_25Site_Stratified...
- sample_data
- AdditionalMasks.tif
- DistanceToEdgeLog2.tif
- FragmentAreaLog10.tif
- HabitatMap.tif
- InvalidAreasMask.tif
- InvalidAreasMask_updated.tif

Code editor content:

```
[13] # Reformat and save to csv
result = pd.DataFrame()
result['longitude'] = long; result['latitude'] = lat
result['row'] = x_strat; result['col'] = y_strat
result['sampled'] = 0
result.to_csv('{}Site_Stratified.csv'.format(ts, nsp), index_label='site')

# Save results as a csv in {} directory \nFile name: {}.format(directory, csv_filename))
results saved as a csv in results/Stratified_Design_Demo directory
2019_11_134437_25Site_Stratified.csv
```

Example output:

You should now be able to navigate to your results folder and open the csv file using your preferred software. We can load and print the first few lines of the output here as a quick check...

```
[14] view_csv = pd.read_csv('{}Site_Stratified.csv'.format(directory, csv_filename))
[15] view_csv.head()
```

	site	longitude	latitude	row	col	sampled
0	1	172.413768	-42.443997	397.0	96.0	0
1	2	172.645018	-42.610305	1019.0	723.0	0
2	3	172.272434	-42.615860	0.0	726.0	0
3	4	172.641166	-42.421514	1022.0	24.0	0
4	5	172.457816	-42.581501	509.0	607.0	0

Once you're happy - test out uploading your own data to generate designs for your study site!